April 14, 2010

Dr. Paul Rispin, Program Manager
Office of Naval Research, Code 331
875 North Randolph Street, Room 273
Arlington, VA 22203-1995

Subject: Deliverable Number 0022, SM21 Modeling, Simulation, and Analysis

Reference: Strategic Mobility 21 Contract N00014-06-C-0060

Dear Paul,

In accordance with the requirements of referenced contract, we are pleased to submit this SM21 Modeling, Simulation, and Analysis

Your comments on this document are welcomed.

Regards,

Dr. John Hwang
Strategic Mobility 21 Principal Investigator

cc: Administrative Contracting Officer (Transmittal Letter only)
         Director, Naval Research Lab (Hardcopy via U.S. Mail)
         Defense Technical Information Center

REPORT DOCUMENTATION PAGE

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
14-04-2010	Technical	Sept. 2007 - April 2010

4. TITLE AND SUBTITLE

Strategic Mobility 21
Modeling, Simulation, and Analysis

5a. CONTRACT NUMBER

N00014-06-C-0060

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Ed Savacool, Program Director
Shui Lan, CoE
Burkhard Englert, CoE

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

California State University, Long Beach Foundation
6300 State University Drive, Suite 220
Long Beach, CA 90815

8. PERFORMING ORGANIZATION REPORT NUMBER

Contractor Report 0022

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Office of Naval Research
875 North Randolph Street, Room 273
Arlington, VA 22203-1995

10. SPONSOR/MONITOR'S ACRONYM(S)

ONR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

CLIN 0022

12. DISTRIBUTION/AVAILABILITY STATEMENT

DTIC Distribution Statement A: Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The SM21-MSA program was created to improve the end-to-end planning for the "delivery" function of both military and commercial supply chains and collaborative regional transportation planning. The SM21 team used both a commercial and academic approach in developing the MSA program. Since a two-pronged approach was used, this document is divided into two sections. The first examines the work completed by the commercial sector and the commercially developed modeling and simulation tools. Examined are reports developed for SM21 based on the TranSystems SCASN model, including the Trade Corridor Gap Analysis and the Simulation Analysis Report. The last section examines the academic side of the SM21-MSA program. The academic program used a dual modeling approach focusing on simulation and optimization, and it used models that are generic, data driven, and flexible.

15. SUBJECT TERMS

Strategic mobility 21 (SM21), Modeling Simulation and Analysis (MSA), Southern California Agile Supply Network (SCASN), Joint Deployment Distribution Support Platform (JDDSP), General Algebraic Modeling System (GAMS),

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. John Hwang
U	U	U		40	19b. TELEPHONE NUMBER *(Include area code)* 1-562-985-7392

Reset

INSTRUCTIONS FOR COMPLETING SF 298

1. REPORT DATE. Full publication date, including day, month, if available. Must cite at least the year and be Year 2000 compliant, e.g. 30-06-1998; xx-06-1998; xx-xx-1998.

2. REPORT TYPE. State the type of report, such as final, technical, interim, memorandum, master's thesis, progress, quarterly, research, special, group study, etc.

3. DATES COVERED. Indicate the time during which the work was performed and the report was written, e.g., Jun 1997 - Jun 1998; 1-10 Jun 1996; May - Nov 1998; Nov 1998.

4. TITLE. Enter title and subtitle with volume number and part number, if applicable. On classified documents, enter the title classification in parentheses.

5a. CONTRACT NUMBER. Enter all contract numbers as they appear in the report, e.g. F33615-86-C-5169.

5b. GRANT NUMBER. Enter all grant numbers as they appear in the report, e.g. AFOSR-82-1234.

5c. PROGRAM ELEMENT NUMBER. Enter all program element numbers as they appear in the report, e.g. 61101A.

5d. PROJECT NUMBER. Enter all project numbers as they appear in the report, e.g. 1F665702D1257; ILIR.

5e. TASK NUMBER. Enter all task numbers as they appear in the report, e.g. 05; RF0330201; T4112.

5f. WORK UNIT NUMBER. Enter all work unit numbers as they appear in the report, e.g. 001; AFAPL30480105.

6. AUTHOR(S). Enter name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. The form of entry is the last name, first name, middle initial, and additional qualifiers separated by commas, e.g. Smith, Richard, J, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES). Self-explanatory.

8. PERFORMING ORGANIZATION REPORT NUMBER. Enter all unique alphanumeric report numbers assigned by the performing organization, e.g. BRL-1234; AFWL-TR-85-4017-Vol-21-PT-2.

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES). Enter the name and address of the organization(s) financially responsible for and monitoring the work.

10. SPONSOR/MONITOR'S ACRONYM(S). Enter, if available, e.g. BRL, ARDEC, NADC.

11. SPONSOR/MONITOR'S REPORT NUMBER(S). Enter report number as assigned by the sponsoring/ monitoring agency, if available, e.g. BRL-TR-829; -215.

12. DISTRIBUTION/AVAILABILITY STATEMENT. Use agency-mandated availability statements to indicate the public availability or distribution limitations of the report. If additional limitations/ restrictions or special markings are indicated, follow agency authorization procedures, e.g. RD/FRD, PROPIN, ITAR, etc. Include copyright information.

13. SUPPLEMENTARY NOTES. Enter information not included elsewhere such as: prepared in cooperation with; translation of; report supersedes; old edition number, etc.

14. ABSTRACT. A brief (approximately 200 words) factual summary of the most significant information.

15. SUBJECT TERMS. Key words or phrases identifying major concepts in the report.

16. SECURITY CLASSIFICATION. Enter security classification in accordance with security classification regulations, e.g. U, C, S, etc. If this form contains classified information, stamp classification level on the top and bottom of this page.

17. LIMITATION OF ABSTRACT. This block must be completed to assign a distribution limitation to the abstract. Enter UU (Unclassified Unlimited) or SAR (Same as Report). An entry in this block is necessary if the abstract is to be limited.

Strategic Mobility 21

Modeling, Simulation, and Analysis

Prepared for:
Office of Naval Research
875 North Randolph Street, Room 273
Arlington, VA 22203-1995

Dr. Paul Rispin, Program Manager
ONR Code 331
703.696.0339
<u>rispinp@onr.navy.mil</u>

In partial fulfillment of the requirements for:
Contract No. N00014-06-C-0060
CLIN 0022

Prepared and Submitted by:

Ed Savacool, Program Director
California State University, Long Beach Foundation
6300 State University Drive, Suite 220 • Long Beach, CA 90815 562.985.7392

April 14, 2010

ABSTRACT

The Strategic Mobility 21 (SM21) team has worked extensively on creating new concepts for improving transportation related network modeling, simulation, and analysis (MSA) capabilities. The SM21-MSA program was created to improve the end-to-end planning for the "delivery" function of both military and commercial supply chains and collaborative regional transportation planning. The SM21 team used both a commercial and academic approach in developing the MSA program. From the commercial sector, previous modeling and simulation work were analyzed, as well as commercially available modeling and simulation programs. The SM21 program enlisted the academic sector to perform experiments using the latest advances in modeling and simulation. The SM21 program maintained a collaborative environment between the academic and commercial teams.

Since a two-pronged approach was used, this document is divided into two sections. The first examines the work completed by the commercial sector and the commercially developed modeling and simulation tools. Examined are reports developed for SM21 based on the TranSystems SCASN model[1], including the Trade Corridor Gap Analysis and the Simulation Analysis Report. The latter validates the SCASN model while also evaluating the JDDSP concept. The Trade Corridor Gap Analysis describes the use of the SCASN Modeling Application for modeling the four major trade corridors within California. The Multi-Modal Terminal model was developed for SM21 by Ablaze Development to design multi-modal terminal operations with a focus on refining the JDDSP concept. The first section concludes by examining the future of military force deployment modeling through the use of a web-based Force Deployment Scheduling Solution. The solution will be developed by the academic program team using AnyLogic, which is a Java programmed, multi-method simulation modeling tool developed by XJ Technologies.

The last section examines the academic side of the SM21-MSA program. The academic program used a dual modeling approach focusing on simulation and optimization, and it used models that are generic, data driven, and flexible. Two modeling subtasks were undertaken to develop modeling capabilities for goods movements through a regional agile supply network. These subtasks were the conversion of the optimization model from MATLAB to GAMS/CPLEX, and a similar conversion of the simulation model from an Arena platform to an AnyLogic based Web Service. MATLAB is useful for small problems with few nodes, but GAMS/CPLEX is better able to efficiently solve more complex problems. The choice of converting from Arena to the AnyLogic Web Service occurred for a similar reason. The Arena platform did not allow for easy integration of the SCASN model into a Service Oriented Architecture; however, the Java-scripted AnyLogic tool allowed for an easy integration through the conversion of SCASN to a web service.

---

[1] The commercial models were created as a family of models built on the same Arena-based simulation platform. The models have the capability of sharing input and output data and dynamic routing algorithms.

Table of Contents

Table of Figures

Modeling, Simulation and Analysis Overview

The Strategic Mobility 21 (SM21) team has worked extensively on creating new concepts for improving transportation related network modeling, simulation, and analysis (MSA) capabilities. The SM21-MSA program was created to support more collaborative regional infrastructure transportation planning and the end-to-end evaluation and optimization for both military and commercial supply chains. The MSA program was also designed to evaluate the effects of the SM21 designed Joint Deployment Distribution Support Platform (JDDSP)[2] on military and commercial transportation networks.

The SM21-MSA program was established using a two pronged development approach: commercial and academic. The commercial sector was contracted to leverage prior work in modeling complex supply chains using commercially available modeling and simulation software. The academic community was engaged to experiment with the latest development techniques in modeling, simulation, and analysis. The SM21 program maintained a collaborative environment between the academic and commercial teams.
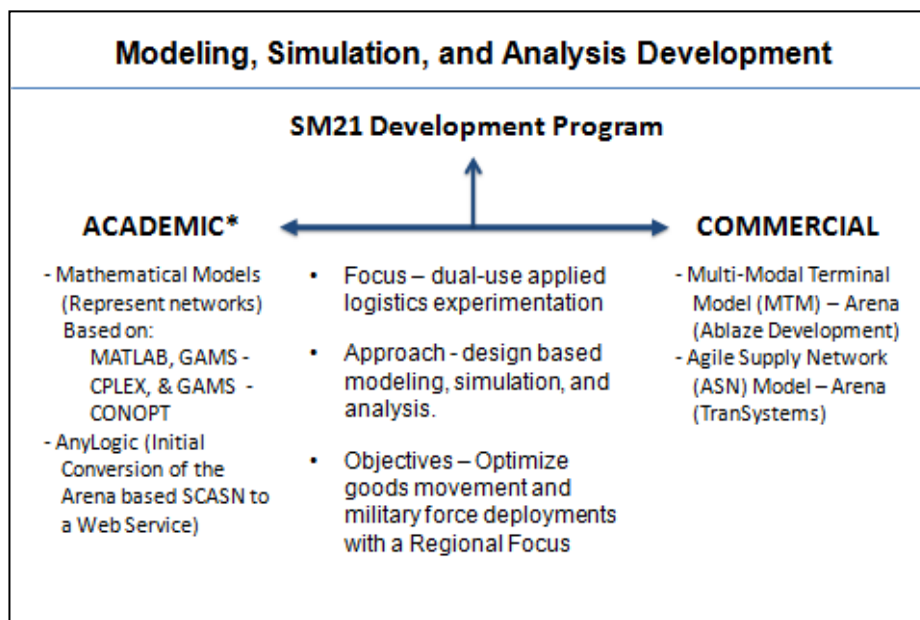
Figure 1: SM21 Modeling, Simulation, and Analysis Development Approach

The high-level overview of the SM21 modeling, simulation, and analysis program is depicted in Figure 1. As noted in Figure 1, the applications developed by SM21 are intended for both commercial and military use. As an example, the SM21-MSA program has supported both the development of a Global Transportation Management System (GTMS) with Dole Foods and the analysis of military force deployment movement schedules from the origin to the port of loading. The MSA program has also analyzed regional transportation planning requirements to include an analysis of the infrastructure associated with the four major California trade corridors.

---

[2] The Joint Deployment and Distribution Support Platform (JDDSP) is a militarily and commercially useful inland multi-modal tra
nsportation node within a distribution network.

Overview of the General SM21 Analysis Approach

For analysis of existing supply chain deliver function networks and regional goods movement networks, SM21 employs the Value Stream Analysis methodology, described below, to determine the as-is state of the network. Once the Value Stream Analysis has been completed, the requirements modeling and specification development efforts are initiated using a variety of tools depending on the network being analyzed.

The Value Stream Analysis methodology employed by SM21 helps to remove "subjective" analysis from the process by quantifying the information and material flow variable (continuous) data at each sequence throughout the supply chain. Even though this requires more knowledge by users, takes longer to gather the data, and the analysis may be more complicated; the advantages include more available information, statistical techniques that can predict trends, and information is provided on the process, rather than just results.

The Value Stream Analysis methodology applied to this effort was a derivation of Value Stream Mapping as pioneered by Womack & Jones of the Lean Enterprise Institute (LEI)[3]. In our initial use of this methodology with Dole Foods, there were over five organizations connected to the Dole Value Stream Analysis Current State, emphasizing both agility (flexibility) and waste reduction. The analysis process selected was designed to eliminate fixing parts of the value stream for each activity without considering the impact on the entire value flow, which often occurs because of capacity constraints, inventories, sales activities, and detours ahead of the next downstream step in the supply chain. The goal was to eliminate changes that result in negligible net cost savings reaching the bottom line, lack of service and quality improvements for customers, lack of systemic long term employee "buy in" into the process, lack of supplier benefits, and limited sustainability as the wasteful norms of the whole value stream close in around the islands of pure value, and frustration all around.

The current state Dole Foods Value Stream Analysis identified the flow of information and material between the "up and down stream" Dole customers. It identified the waste and value in current processes and enabled Dole employees to think in terms of:

- Processes, Not Products or Functions
- Mapping Flow of Materials and Information between the Dole "up and down stream" internal and external customers
- Value Creation & Not Price
- Maximizing Supply Chain Relationships & Security
- Waste Removal Through Continuous Improvement
- Assigning Costs to Activities (not resources)

Given the supply chain cost drivers, SM21 found the Value Stream Analysis provides a proven waste reduction strategy which includes measuring Current State unit input per unit of output for all resources including energy. The Value Stream Analysis Future State then designed Kaizens

---

[3] Value Stream Mapping principles described in this report are excerpted from "Learning To See" written by James Womack & Dan Jones of the Lean Enterprise Institute (LEI).

(process optimizations) for an improved Future State to help drive waste out of the Dole supply chain.

The models and simulations developed by SM21 are employed to design the process optimizations. The following sections describe the models and simulations designed and developed by SM21 to support the Value Stream Analysis and complete the deliver function and regional goods movement evaluations and optimizations.

Section I: SM21 Commercial Approach to Model Development

Overview

As previously noted, the SM21-MSA program was developed through a two pronged commercial and academic development approach. The commercial approach leveraged the experience gained by commercial corporations in modeling complex supply chain distribution networks. As an example, the TranSystems' Transportation Modeling Studio ^{TM} which enables planning-level capacity analysis for marine and intermodal terminal facilities, regional rail systems and surface transportation networks was used to develop the Arena[4] based SCASN modeling and simulation system.

Figure 2: The Southern California Agile Supply Network Modeling System

---

[4] Arena is a discrete event simulation software simulation and automation software distributed by Rockwell Automation

As depicted in Figure 2, the initial SM21 Southern California Agile Supply Network, or SCASN, model was built by TranSystems using Arena simulation software and the Transportation Modeling Studio^{TM}. The SCASN modeling and simulation program was designed to be generic in nature to allow for use by both commercial and military entities. As part of the model creation and testing process, SM21 has developed and released several reports and documents related to the modeling and simulation program, which are listed as references in Appendix B and overviewed below.

Trade Corridor Gap Analysis

As part of the current development under the Office of Naval Research sponsored SM21 program, two analyses were completed using the TranSystems developed SCASN model. One was a gap analysis of trade corridor in Southern California, the original focus of the SCASN model. The report, *Trade Corridor Gap Analysis*[5], uses the SCASN model to show how changes in infrastructure capacity and operating strategy can affect the flow of traffic throughout the region of interest. The document describes how the highway network and truck movements can be modeled with the SCASN program, and it provides an example of a simulation and the resulting data.

The report describes how by employing the SCASN modeling program, the user can analyze a wide range of data. The data can be used to determine the impact of dedicated truck lanes, the addition of highway lanes, or the addition of new highways. The program also gives the user the ability to see and analyze the effects of changes in transportation management policies and practices. The report offers suggestions to improve the SCASN model, such as a method to increase the accuracy of traffic flow. It also suggests including the ability to track elements that impact truck movements, such as fuel and the environment.

Military Simulation Analysis Reports

The second analysis report is the *Simulation Analysis Report*[6]. This document analyzes the modeling and simulation performed by TranSystems under the SM21 project to internally verify and validate the SCASN model while analyzing the JDDSP concept. A high-level overview of the SM21 JDDSP development effort is depicted in Figure 3. The modeling for the JDDSP was completed using the SCASN simulation platform. While it can be used for commercial purposes, the SCASN model was originally developed and validated using military data.

This document looks at four different simulation scenarios with the purpose of proving SCASN's capability to model scenarios that originate outside of the Southern California Areas and terminating at the Ports of Long Beach and Los Angeles. Two of the simulations were Sea-basing[7] scenarios, one using Victorville as the JDDSP, and one based on current distribution

[5] TranSystems Corporation, *Trade Corridor Gap Analysis*, Strategic Mobility 21, July 31, 2008

[6] TranSystems Corporation , *Simulation Analysis Report*, Strategic Mobility 21, July 31, 2008

[7] Seabasing – from an SM21 perspective includes the rapid deployment and re-employment of joint combat power from the sea, while providing continuous sustainment to select expeditionary joint forces without reliance on land bases within the Joint Operational Area (JOA).

practice without employing the JDDSP. Likewise, two Mechanized Combat Brigade Force Deployments Scenarios were analyzed, one based on current deployment practices and one with the inclusion of the JDDSP as a node within the Joint Deployment and Distribution Enterprise (JDDE)[8]. The simulations were meant to determine the scheduled for units to leave home stations in order to meet scheduled ship calls at the strategic port. The schedules were subsequently used to minimize the buildup of equipment in commercial ports prior to the start of ship loading operations. The Sea-basing scenarios allowed the user to develop the equipment movement schedules and also evaluate the buildup of cargo at the JDDSP and strategic port of loading.
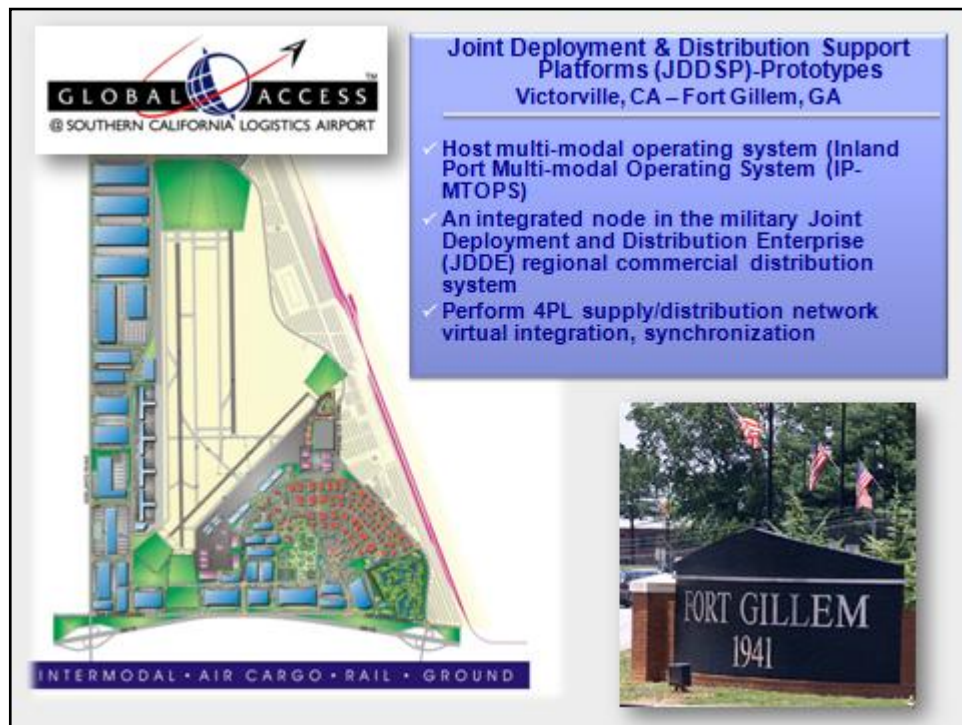
Figure 3: The Joint Deployment and Distribution Support Platform Overview

Multi-Modal Terminal Model

The SM21 team created a separate Arena application to model the Multi-Modal Terminal within the JDDSP, described in the document *SM21 Multi-Modal Terminal Model Documentation*[9]. This model, called the Multi-Modal Terminal (MMT) Model, was originally developed using Arena Basic v 10.00 and was updated in 2008 to allow for incorporation with TranSystems' developed SM21 SCASN models. The MMT model contains five sub-models. They are the

---

[8] Within the context of this report, the Joint Deployment and Distribution Enterprise (JDDE) refers to a multi-modal, in-land transportation node including the complex of procedures, technical connectivity, and information, necessary to conduct joint distribution operations.

[9] *Multi-Modal Terminal Model Documentation,* Strategic Mobility 21, Ablaze Development Corp, Version 2, July 2008

Highway Activity sub-model, Airlift Activity sub-model, Rail Activity sub-model, Storage Activity sub-model, and the Vessel Activity sub-model. The document provides a brief description of each sub-model. The document also gives detailed instructions on using the model program, including adjusting various configurations. The various reports available for review after the completion of a simulation are provided in this document as well. These reports included a Resources Detail Summary and a Queue Summary, both of which are used as an example in the document, along with others.

Agile Deployment Process Modeling and Simulation

For the past seven years, the Center for the Commercial Deployment of Transportation Technologies (CCDoTT) has supported the military analysis associated with the dual-use Agile Port System (APS). SM21 has supported the analysis over the past three years and is now planning the development of the information management system that would enable agile deployment processes required by the APS. TranSystems has also supported the APS program and has developed all of the APS models and simulations. The CCDoTT and SM21 programs have jointly developed a revised modeling and simulation approach to enable the military force deployment processes associated with the APS. The following sections provide an overview of a proposed web-based force deployment scheduling solution is under development by the CSULB College of Engineering. Section II contains additional information on the development of the deployment scheduling service using the AnyLogic modeling and simulation application.

Simulation-based Systems Planning for Force Deployment Support

Based on the experience gained in evaluating the military force deployment networks and processes, TranSystems was asked to provide a White Paper describing how supply chain networks, transportation facilities, and military force deployment networks utilize a combination of infrastructure, assets, contracts, and processes to accomplish shipment goals. The paper defines that often changes in more than one of these elements provide the best overall solution. Systems can be complex and have many interactions such that it is difficult to really determine "What should be done?" for planning and management of supply chains and military force deployments. The focal point of the White Paper was to present, because of the variables associated with force deployments, the best use of a simulation model to provide the ability to iteratively analyze what should be done relative to what can be done:
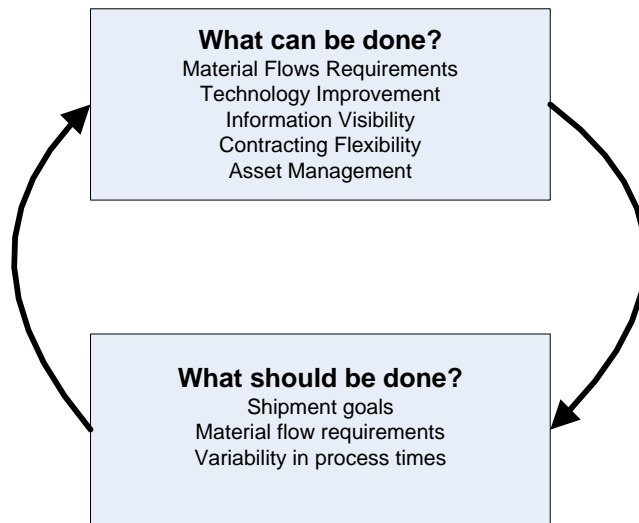
What can be done?
Material Flows Requirements
Technology Improvement
Information Visibility
Contracting Flexibility
Asset Management

What should be done?
Shipment goals
Material flow requirements
Variability in process times

Figure 4: Systems Planning Process

The paper describes how it is critical to look at supply chain and force deployment networks or set of transportation facilities as a "system" to determine what combination of changes are needed to support shipment goals. A modeling approach that considers a "plan for success", which in the context of a force deployment considers getting to an outbound vessel "a schedule determined order" to meet stow plan requirements.

The SM21 program and CCDoTT jointly developed a prototype model framework that "backwards schedules" the military equipment loading process onto a strategic sealift ship as the "first planning step" in developing the "plan for success"[10]. The modeling will include all the various material and equipment flows and how they collectively or independently use network facilities and resources. Each material flow has its own process breakdown of the steps needed from "fort to port".

The force deployment modeling developed is time-based and allows for the impact of concurrent use of resources to be included where needed. These resource constraints can impact the ability of goods and equipment to make their destinations as needed. Other analysis approaches are valuable to determine how the overall network might be improved (i.e. Value Stream Mapping[11]) —simulation adds the element of time to assist in determining how concurrent material and equipment flows that need to use the same resources impact each other. For military force deployments, it is not enough to consider only the military assets, facilities, and processes. The interaction of the military deployment with the entire commercial and public transportation network must be considered. This provides a system level perspective to determine what level of infrastructure, process, or other change is really necessary to meet force deployment requirements.

---

[10] In terms of Joint force deployments, successful plans support the Combatant Commander's force deployment and employment intent.

[11] As defined by SM21, value stream mapping is a lean manufacturing process used to analyze the flow of materials and information currently required to bring a product or military unit to the point of need.

Web-based Force Deployment Scheduling Solution

The SM21 MSA program academic team is currently developing a web-based, highly flexible model that includes the ability to change all system elements to provide the ability to assess overall force deployment performance and goals. A three-stage modeling approach was suggested by TranSystems, which has been adapted by SM21. As a first stage, modeling will be used to determine what the preferred deployment plan is given force deployment process designs, commercial and military facility constraints, contracts, etc. At this stage, the system design will be iteratively adjusted as necessary to meet the overall plan. The system design might identify what type of visibility or information exchange is necessary between the different process "nodes" or operating entities (commercial or military). This may identify areas where improved information exchange is critical to meet overall objectives.

A second stage of force deployment modeling will provide a schedule of material movement requirements (release dates) based on the preferred system design identified during Stage 1. This will be a web-based analysis supported by the AnyLogic[12] and if there are real-world changes to the needed release times, these changes can be added and a "replanning" can provide an updated schedule for downstream processes to compensate for the change or identify a potential delay in ship loading for planning purposes. A late ship loading time would, in most cases, flag the need to determine if there are some alternative process changes that could be made to meet system goals.

As a stage 3 type of analysis, and if a new process change is required, the model can be used to test and determine which option will best solve the problem. Once the best alternative decision is determined, a new schedule can be generated for management of material and military equipment movements.
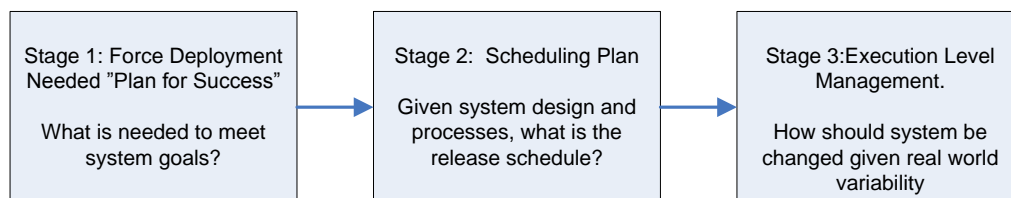
Figure 5: Multi-Stage Force Deployment Modeling Approach

The multi-staged, web based force deployment modeling solution provides several benefits:

1. Provides planning level insight as to how a force deployment network, supply chain network or set of transportation facilities needs to be designed and operated.
2. Provides a management tool to provide release schedules.
3. Allows for real-world variation to be incorporated and the ability to "re-plan" and try alternative tactics to attain system shipment goals.

---

[12] AnyLogic is a multi-method simulation modeling tool developed by XJ Technologies.

SM21 Model and Simulation User Guides

The SM21 Team has produced two documents to help the user when using the SCASN Arena model. The first is the *Southern California Agile Supply Network Simulation User's Guide Manual*, and the second is the *SCASN Getting Started Checklist*. The *User's Guide* provides an overview of the SCASN model and the system requirements needed to run the model. The manual describes how to create a master scenario, how to input data, how to run the simulation, and how to view output data. The guide provides information on each of the main tabs and controls that a user will come across and utilize when using the SCASN modeling program. This guide states that the modeling program is designed to be flexible and generic for use in any region or agile network scenario.

The *SCASN Getting Started Checklist* reiterates the generic and flexible design of the SCASN model and seeks to clarify the problem caused by these characteristics. One problem with the generic nature of the model is with the meaning of different terms. The checklist defines two important terms: schedule service which includes trains; and, outbound processing, which includes trucks. This document also provides a glossary to help the user understand the meaning of the terms used in the SCASN simulator. Finally, the checklist presents eight (8) steps for using the simulator. The steps are:

1) Define facilities, shipment assets and freight types
2) Define the feasible flows between facilities
3) Determine the timing profile of each flow
4) Set outbound distribution
5) Specify the valid interface flows
6) Create the originating volume
7) Define scheduled services
8) Run model and verify flow.

The document provides advice and questions for each step to guide the user before moving onto the next step. This checklist created by the SM21 team ensures that the user can correctly run a simulation using the SCASN modeling program.

Section I Summary

The SM21 team will transition the models and simulations developed by the commercial development team for continued use and further development by the commercial and military communities of interest. SM21 has produced several detailed documents related to the development of the simulation and modeling efforts. These efforts included two analysis and two documents related to the use of the simulation and modeling program. A fifth SM21 document describes the development and use of the Multi-modal Terminal (MMT) model. A sixth document provides the foundations for the development of a web-based SCASN model and simulation that will support force deployment operations and other distribution network analysis. These documents show the progression of the development, use, and current transition of the SCASN model. The following section provides an overview of the Academic approach to the SM21 model and simulation development program.

Section II: SM21 Academic Modeling and Simulation Support

Migration of Optimization from Matlab to GAMS/CPLEX and of Simulation from Arena to an AnyLogic Web Service

Shui Lam and Burkhard Englert

Section II Executive Summary

The purpose of the modeling subtasks is to develop modeling capabilities for goods movements through a regional agile supply network. The models developed are generic and data driven. A specific network is defined with input parameters so that the underlying network can be changed "on the fly." They can be applied to all agile supply networks that consist of some or all of the following components involved in goods movements: a container port, regional intermodal terminals, inland terminals, regional distribution centers, local warehouses for major importer/exporters, and an underlying regional transportation infrastructure. Two regional modes of transportation are considered: road, and rail. The Southern California Agile Supply Network (SCASN) is used as an initial implementation.

We follow a dual approach for our modeling effort: optimization and simulation. While it may seem that many questions can be answered directly through simulation, we believe that a network optimization can provide very valuable additional information. Our goal with this dual approach is to address the following research questions:

- What are the most efficient and effective freight mobility strategies to transport containers (local and national distribution)?
- Can the optimal strategies be reasonably realized? If not, where are the bottlenecks?
- What is the net throughput impact of insertion of an inland multimodal transfer facility (Agile Port) on the system? Where is its optimal location?
- What are the impacts of joint force deployment on commercial freight mobility and infrastructure capacity use?
- Do capability gaps exist under various "what if" scenarios? With these logistic analyses, we will be able to determine the costs and benefits of infrastructure changes, policy or business rules changes.

To achieve these goals we have constructed a basic mathematical formulation of a network flow model and implemented this model using MATLAB. While MATLAB provides sufficient capabilities to solve limited size model with only few nodes its performance degrades rapidly once the size of the model increases. For this reason we converted the implementation of our network optimization model from MATLAB to GAMS / CPLEX. We applied the optimization model to the Southern California region. We have also formulated the Dole shipment scheduling problem as an optimization problem and developed an optimization model for its solution using GAMS / CONOPT. Furthermore, we developed a simulation model of the agile supply network

using ARENA considering random arrivals of containers and delays due to congestion on roadways, and later converted the ARENA simulation model to the AnyLogic Platform.

AnyLogic is Java-based and hence allows us to make supply chain network simulations available remotely as a Web Service. Such a Web Service can be incorporated into a Service Oriented Architecture (SOA). Our AnyLogic Web Service currently has limited capabilities. It allows a user through a browser window to pause and restart a simulation that is running on a server. We have also developed a tool that converts network definition data input on Excel spread sheet into a SQL database. The Web Service implementation, however, is not yet completed. We need to add functionalities that allow the user to control the simulation run and modify its execution. Also the user should be provided with the ability to upload a new network and new network data and run a simulation based on this data. Finally, the user interface of our Web Service should be improved and the web Service integrated into the SOA.

Modeling and Simulation

The purpose of the modeling subtasks is to develop modeling capabilities for goods movements through a regional agile supply network. The models developed are generic and data driven. A specific network is defined with input parameters so that the underlying network can be changed "on the fly". They can be applied to all agile supply networks that consist of some or all of the following components involved in goods movements: a container port, regional intermodal terminals, inland terminals, regional distribution centers, local warehouses for major importer/exporters, and an underlying regional transportation infrastructure. Two regional modes of transportation are considered: road, and rail. The Southern California Agile Supply Network (SCASN) is used as an initial implementation.

Southern California Agile Supply Network

The Southern California Agile Supply Network (SCASN) is a complex transportation network where nodes represent source, destination, and transfer facilities and arcs represent rail and major surface roads.

Our SCASN model:

- Provides both user reconfigurable nodes and segments.
- Supports user defined constraints, restraints and environments.
- Permits optimization of single and multi-dimensional objectives.
- Will be accessible, usable and reconfigurable via a Service Oriented Architecture (SOA).
- Is employed across multiple transportation modes.
- Can integrate "single point" node or segment models.

The academic team adopted a dual approach in our modeling effort by first using optimization techniques to examine various policy decisions such as multimodal transport and network design, and in a second step simulation techniques were employed to identify current and future bottlenecks and investigate effects due to randomness. Cost and benefit tradeoffs for are provided for each option evaluated.

As a result of the dual modeling effort – optimization and simulation – the following research questions can be addressed:

- What are the most efficient and effective freight mobility strategies to transport containers (local and national distribution)?
- Can the optimal strategies be reasonably realized? If not, where are the bottlenecks?
- What is the net throughout impact of insertion of an inland multimodal transfer facility (Agile Port) on the system? What is its optimal location?
- What are the impacts of joint force deployment on commercial freight mobility and infrastructure capacity use?
- Do capability gaps exist under various "what if" scenarios? With these logistic analyses, we will be able to determine the costs and benefits of infrastructure changes, policy or business rules changes.

The SCASN model developed by the academic team is generic. A specific network is defined with input parameters so that the underlying network can be changed "on the fly". The model was applied to the Southern California regional network but it can be easily extended to other networks, such as the Savannah Port region. The national transportation network can be evaluated with the model with an appropriate degree of aggregation.
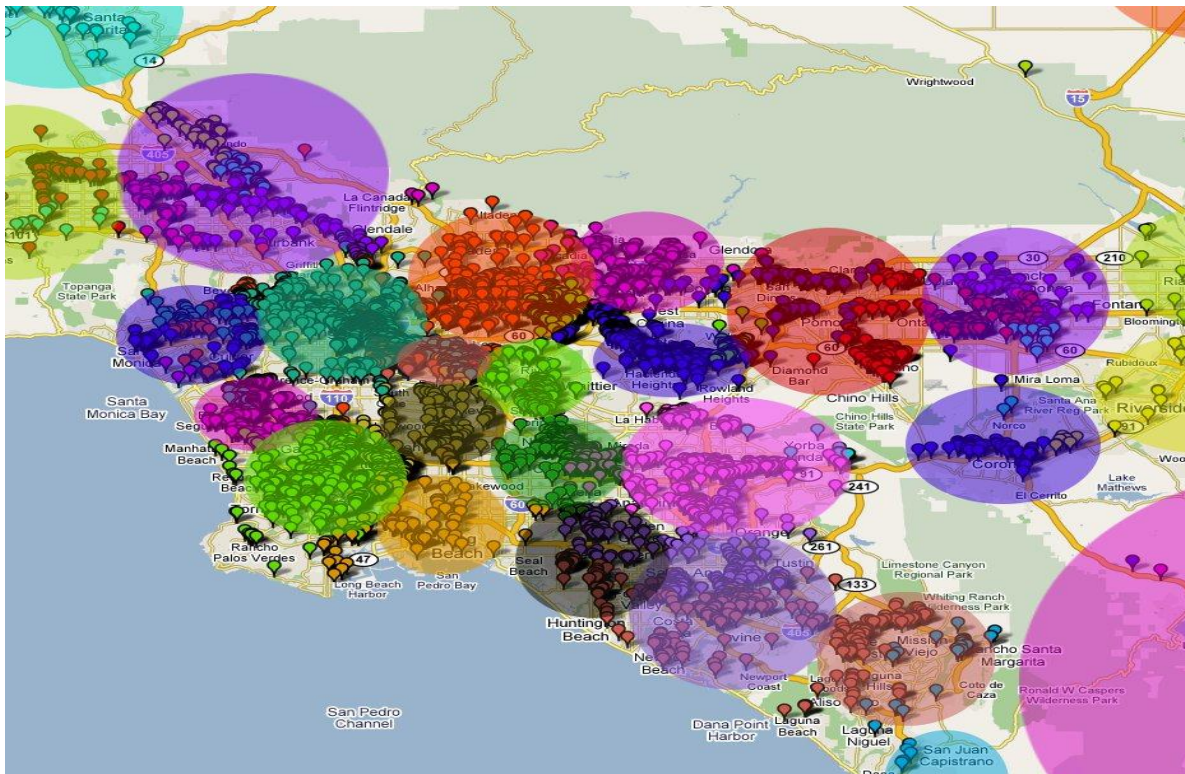
Figure 6: Warehouse Locations and Aggregations in the Southern California Region

To ensure the feasibility of the optimization and simulation models, aggregation may be necessary. Figure 6, for example shows how warehouse locations in the Southern California region could be aggregated to reduce the complexity of the system allowing for an efficient solution. The level of aggregation needed depends on the available computational resources and any possible time restrictions. With more resources and less time restrictions less aggregation is needed, increasing the fidelity of the outcome. Overall our model is flexible - it leaves it to the user to define a network with the desired level of aggregation.

Dual Approach – Optimization and Simulation

In our modeling effort we followed a dual approach, Optimization and Simulation. While it may seem that many questions can be answered directly through simulation, we believe that a network optimization can provide very valuable additional information.

Optimization and Simulation

We use *optimization* because it allows us to compare the current practice with an ideal (optimal) scenario. Optimization tells us whether there is room for improvement and if the expected improvement is significant enough to warrant an investment into a change in business process or an infrastructure addition. Many international trade and freight transportation problems have been investigated with optimization modeling and found cost saving alternatives. Here are some examples of applications of optimization in supply chain management:

- Goods movement in regional transportation network
- Ship loading/unloading sequencing
- Export/import yard configuration
- Empty container reusing
- Warehouse layouts
- Goods delivery sequencing
- Several others

Once an optimal solution to a given problem has been found, *simulation* can be used to verify whether this optimal solution is feasible and what preconditions and consequences its implementation would have. Simulation takes into consideration business processes, random occurrence and sequencing of events, congestions, and so on. If the optimal solution is feasible under these conditions, the simulation provides a workable realization of the optimal solution. Otherwise, it helps identify bottlenecks in the network and point out improvements needed in order to achieve the optimal or near-optimal result.

Solving Optimization Problems

An optimization problem is formulated aiming to finding the best way to achieve an objective in terms of a cost function. These types of problems typically are straightforward to solve when the problem is small, but become extremely hard due to the amount of computations required. This phenomenon can be illustrated with a simple problem given below.

Let us assume that goods are packed in a container (or delivery truck) at a distribution center (DC) and are to be delivered to a number of customers in a metro area. The delivery route needs to be mostly planned out in advance in order to pack the container properly to avoid major reshuffling. Also, different routes will have different costs due to a difference in total distance traveled and time taken. So the question arises which route is least cost?

We can model this problem as the classic "Traveling Salesman Problem" (TSP) in which a least-cost route is to be identified for the salesman to start from his home city to visit every city of the set of cities within his responsibility exactly once for sales promotion. We can also apply the TSP to the delivery routing problem by modeling the truck as the salesman, the DC as his home city, and all delivery points as the cities he needs to visit. This is modeled as an optimization problem. The objective is to minimize the total cost, based on a cost function we want to define (e.g., distance traveled, travel time, or monetary cost that reflects all relevant factors including distance, time, etc.). The solution must also satisfy a set of constraints, including: all delivery points must be reached exactly once, travel must start and end at the DC.

The distance and/or travel time between each pair of locations are given. A straightforward solution strategy simply examines all possible routes, computes the cost of each, and identifies the route with the least cost. If we have only a few delivery points, say for example 3, the solution is simple. Only 3!, or 6 routes need to be examined. This can easily be done by hand. However, the number of possible routes increases extremely fast – n!/2 for a set of n delivery points. This phenomenon, i.e., that the complexity increases extremely fast (as an exponential or factorial function) as problem size increases, is common to many optimization problems.

Good and efficient optimization algorithms not only look for the optimal solution to a given problem but they also optimize the search for this optimal solution. As a result we looked for options that implement this optimization. One such option is MATLAB.

MATLAB

Initially we implemented our constructed optimization models in MATLAB. MATLAB is a numerical computing environment and fourth generation programming language. Developed by MathWorks, MATLAB allows matrix manipulation, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs in other languages. While MATLAB provides sufficient capabilities to solve a limited size model with only few nodes, its performance degrades rapidly once the size of the model increases. In recent years the CPLEX optimizer became known as one of the best algorithms capable of optimizing its own execution and search. CPLEX runs in the General Algebraic Modeling System (GAMS). For this reason we converted the implementation of our network optimization model from MATLAB to GAMS / CPLEX.

GAMS/CPLEX

To solve optimization problems, and in particular to solve them efficiently, it is critical that the examination of possible solutions is done efficiently. For this purpose we use an optimization tool that optimizes the implementation of its own optimization algorithms. Such an optimization of implementation can usually be done much more efficiently automatically than by hand.

It is crucial that the modeling language allows users to precisely define the underlying mathematical structure of the given problem. At the same time the modeling language should be as high level and accessible as possible. This is why CPLEX uses the General Algebraic Modeling System (GAMS). GAMS was specifically developed to provide a high level language

for the compact representation of large and complex models. It allows changes to be made in model specifications simply and safely. Algebraic relationships can be expressed unambiguously and model descriptions are independent of solution algorithms.

GAMS

GAMS was designed to incorporate ideas from relational database theory and mathematical programming and merges ideas from these areas to assist strategic modelers. Relational database theory provides the framework for developing general data organization and transformation in GAMS, and mathematical programming provides ways to describe problems and the methods to solve these problems. GAMS has the following properties:

- All embedded algorithmic methods are available without changing a user's model representation. New methods or modified implementations can be done without making changes in existing models. The system currently has capabilities for linear, nonlinear, mixed integer, mixed integer nonlinear optimizations, and mixed complementary problems.
- The optimization problem is expressible independent of the data it uses. This separation of logic and data allows the size of the problem to increase without having to increase the complexity of the representation.
- To enable the use of a relational data model, computer resources are allocated automatically. Users do not need to worry about details such as array size and storage problems.

Other useful features of GAMS include:

- Program self-documenting: Models are represented in a manner that is both easily readable by computers and humans. The GAMS program itself is the documentation for the user and separate documentations are not required. The model representation furthermore is concise and uses the efficiency of the underlying mathematical representation. All data transformations are specified algebraically. Data can be entered in its most basic form, and all transformations of data are available for inspection. Explanatory text can be made part of the definition of symbols and is then reproduced anytime the corresponding values are displayed. The model representation itself is in a single document.
- Model portability: Models can be solved on different types of computers with no change. A model can be developed on a small laptop or PC and later be solved on a large mainframe. Models can be developed in one location and be used anywhere else, only the GAMS statement of the model must be moved. This statement of the model contains all logical and data specifications that are needed to solve the model. This is particularly useful in the context of SM 21.
- Support user choice of editor: GAMS does not require the use of a specific editor. Every user can use the editor or word processor of his or her choice. GAMS can hence be integrated with existing user environments.
- A useful feature of GAMS is its existing model library together with a database to help locate examples that cover countries, sectors, or topics of interest.

- GAMS Data Exchange (GDX): A GAMS Data Exchange (GDX) file stores the data used by a GAMS model and the data produced by a GAMS model. GDX files are binary files that are portable between different platforms. Moreover, GAMS includes the GDXXRW utility that allows converting Excel spreadsheet data to and from a GDX file. This greatly facilitates our ability to analyze the data produced.

CPLEX

GAMS/Cplex is a solver that allows the combination of the modeling capabilities of GAMS with the power of Cplex optimizers. Cplex optimizers are designed to solve large problems quickly and with minimal user intervention. GAMS/Cplex provides several solving options which are automatically calculated and set at best values for specific problems. Cplex can solve problems using linear programming, quadratically constrained programming or mixed-integer programming. For our applications we will mostly use Cplex's linear programming capabilities.

Cplex can solve linear programming problems using several alternative algorithms. In most cases the dual simplex algorithm is used. The primal simplex algorithm, the network optimizer, the barrier algorithm and the sifting algorithm are also provided. With an appropriate license, the concurrent option allows solving with different algorithms in parallel. The system then returns the result of the first algorithm to finish.

Linear programming algorithms are very memory intensive. In cases where memory is limited Cplex automatically makes adjustments to allow execution.

Modeling Effort

Our GAMS optimization model includes arcs that represent surface roads as well as arcs that represent rail links. We are also able to represent intermodal facilities where goods are moved from truck to rail or vice versa. This allows us to develop best case solutions that include the use of road as well as rail, and any other types of transportation links.

We completed the conversion of our least cost network flow model from MATLAB to GAMS, and tested our implementation using a combination of real and estimated input data for an agile supply network. We also applied our GAMS/CPLEX optimization algorithm to solve an optimization problem in the Dole Foods case study as part of the Dole Experimentation.

The problem at hand is to compare three Southern California warehouse locations: Buena Park (current location), SCLA, and near port. The comparison is based on the distribution cost to all customers serviced from there, using Dole delivery data for the demand pattern. It is clear that this is an optimization problem that can be solved directly with the CPLEX model we developed for SCASN. A model was set up by matching Dole delivery data onto the SCASN node-arc network and using various cost parameters (incl. drayage, occupancy, labor, and transportation). We ran the model three times for the comparison, each with one warehouse location in consideration, using the 2008 YTD volume. Specific data used for the study are given as follows:

- Volume of distribution: 5652 containers, by truck. Breakdown of within vs. out of region: 1873 vs. 3779.
- Drayage cost per container
  - To Buena Park: $170
  - To Long Beach (near port location): $145
  - To SCLA: $400
- Occupancy and labor cost per container
  - At Buena Park: $682
  - At Long Beach: $682
  - At SCLA: $358 (not counting the one-time incentive)

The experimentation results show that the total distribution costs from the 3 locations are ranked as follows (from lowest to highest):

- Buena Park: $7,672,431
- Long Beach (near port): $8,456,957
- SCLA: $8,638,845 (not counting 1-time incentive, 12.6% higher than Buena Park), or SCLA: $8,155,095 (includes 1-time incentive, 6.3% higher)

Based on the selected cost factors, the volume, and demand parameters, the current warehouse location is the best among the 3 considered. To ensure the reliability of this result, one should perform sensitivity analysis to determine robustness of models, as well as develop an end to end simulation of the business process that includes all the relevant data.

Shipment Scheduling Optimization for Dole Foods

We applied the optimization modeling methodology for a specific container shipment scheduling problems for Dole Foods as part of the Dole Experimentation. Dole Foods produces their products in some parts of the world, packages them into containers, and then ship to the United States for distribution. For example, containers originated from South China or Southeast Asia may be loaded onto container ships at major ports in Asia like Hong Kong and Yan Tian, China for the ports of Los Angeles or Long Beach in Southern California. From the Twin Ports, containers are either transported via rail service to a warehouse out of the region or picked up by trucks for a local warehouse where further distribution will take place. We will focus on the problem that involves local truck pickups in this study that attempts to find an optimal plan of shipments of the containers over a given planning horizon by considering trade-offs between shipping costs and free dwell time at the destination terminals.

Upon a shipment's arrival at a marine terminal in the destination port, the containers are unloaded onto the terminal ground and the shipper of the goods will be notified for their pickup. Marine terminals typically give a few days of free storage allowance, beyond which a demurrage charge per day per container will be levied. Shippers often contract multiple shipping lines, which charge different shipping rates and provide different free dwell time allowance. Each shipping line maintains a different voyage schedule. The number of containers shipped in a given shipment must be sufficient to meet the cumulative demands until the next shipment's arrival. The number of containers that can be transported out of the terminal is limited by the capacity of the warehouse and the transportation service the shipper employs. Therefore, it will

take time for the shipper to clear all the containers from previous shipments, and it is conceivable that, if not planned well, the amount of dwell time of their containers will add up and demurrage cost will incur. With multiple carriers involved and each charging different shipping rate and having different demurrage charge policy, the decision on an optimal shipment schedule could be very complex. We have developed a mathematical programming model to help find such an optimal container shipment schedule from the point of origins to the warehouse destination within a given planning horizon. The optimality is in terms of end to end cost.

Modeling the demurrage costs

A shipment of containers arriving on the carrier will be unloaded onto the terminal. The shipper is notified of the containers' arrival. The shipper will then arrange for the containers' pickup and their delivery to a warehouse location. Marine terminals typically allow a limited number of days of free storage for unloaded containers. Beyond the free period demurrage charge will be imposed. Containers arriving in the next shipment will add to those that are still remaining in the terminal at that time. To minimize the demurrage charge, the shipper would obviously move the oldest ones out first, Moreover, the shipper must schedule the shipments in such a way that they will not exceed the transportation capacity to the extent that container inventory in the terminal will continue to build up.

To study the problem we define the following notations using a simple scenario that involves a single carrier and only one type of containers is to be transported.

a_n : Number of containers that arrive at the destination port on Day n

t_n : Number of containers picked up and transported off the terminal on Day n

s_{nh} : Number of containers arrived on Day n that have been kept in the terminal for h days

p_n : Number of containers that incur demurrage charge on Day n

F : Number of free storage days provided by the terminal

We are particularly interested in how many containers that arrive on day n are still in the terminal at day $n + F$. These are the containers that incur demurrage charges. To find out how many such containers will incur demurrage charges we must carefully keep track of all the containers that arrive and determine when they will leave the terminal. We assume that containers are picked up from the terminal in the order in which they were received. That is to minimize costs the shipper will always first pick up the "oldest" containers first. Clearly, in the real world, a customer may want to prioritize certain containers, that is potentially pick them up as soon as they arrive (are cleared for pick up). We assume, however, that prioritized containers can be tracked separately from the rest of the shipment, and the model parameters can be set to reflect the problem with the prioritized containers excluded. With no priority involved, a reasonable approach to minimize overall transportation cost would be to keep demurrage cost low by moving containers out of the terminal as soon as trucking and warehouse capacities allow, i.e., using a First-In-First-Out (FIFO) approach. Developing a more comprehensive model that reflects prioritization will be future work.

Our goal is to minimize cost. To do so, a shipper must schedule shipments and pick-ups from the terminal so that overall shipping cost is minimized. Naturally a shipper will want to avoid demurrage charges. Without demurrage charges our model would be linear. Since the demurrage charges, however, are not constant but depend on how many containers from earlier shipments remain at the ports, which in turn depends on how many containers were shipped earlier, our problem becomes non-linear. To obtain a solution it is key to compute the number of containers that incur demurrage charges as a result of a given shipping schedule. This requires some careful bookkeeping with respect to the length of time containers remain at a terminal.

To keep track of all the containers that incur demurrage charges we use the variable s. Of the arrivals a_1 on Day 1, a portion, t_1, will be transported away to the warehouse. Those that are not picked up will remain in the terminal. The number of such containers, $s_{11} = a_1 - t_1$ as depicted in Figure 7 below, where the length the box represents the size of the arrivals.
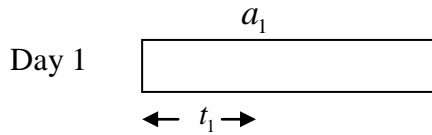

Figure 7: Containers in Terminal on Day 1

On Day 2, a_2 containers arrive and t_2 are trucked away. With the s_{11} containers left behind from Day 1, the number of containers that will stay behind in the terminal depends on the amount of t_2 relative to s_{11}. We illustrate the two possible cases in Figure 7. Here the total length of the two boxes represents all containers in the terminal that include the new arrivals a_2 and those left behind from Day 1, s_{11}.
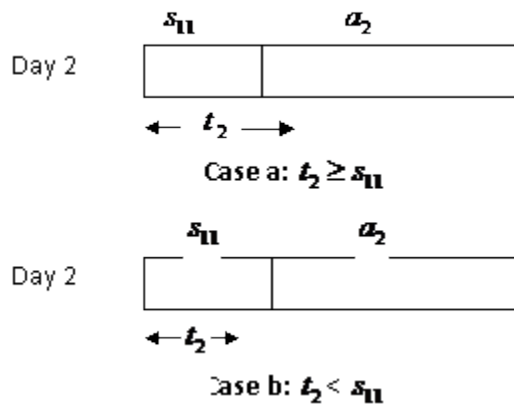

Figure 8: Containers in Terminal on Day 2

If $t_2 \geq s_{11}$, as depicted in Case a in Figure 8, $s_{21} = a_2 + s_{11} - t_2$ whereas $s_{12} = 0$ since the pickups will deplete all those arrived on Day 1 plus some from the Day 2 arrivals, based on our FIFO assumption. If on the other hand $t_2 < s_{11}$ as depicted in Case b in Figure 8, then Day 1 arrivals

will cover all demands on this day with some remaining in the terminal for a second day as computed by $s_{12} = s_{11} - t_2$, while all the Day 2 arrivals will be left in the terminal, yielding $s_{21} = a_2$. Combining these two scenarios the calculation for the number of containers that will be left behind on Day 2 after the pickups will be defined as below, with s_{12} representing the number left in the terminal from Day 1 arrivals for a second day and s_{21} the number from Day 2 arrivals staying for the first day. Note: s_{12} may be zero.

$$s_{12} = \max(0, s_{11} - t_2)$$
$$s_{21} = a_2 + s_{11} - t_2 - s_{12} = a_2 + (s_{11} - s_{12}) - t_2$$

On Day 3, a_3 containers arrive and t_3 are trucked away. With the s_{12} and s_{21} containers left behind from Day 1 and Day 2, respectively, the number of containers that will stay behind in the terminal after Day 3's pick-ups depends on the amount of t_3 relative to s_{12} and s_{21}. To determine the number of containers that will remain in the terminal after pick-ups, we observe that there are three possible cases: Case a where $t_3 \geq s_{12} + s_{21}$; Case b where $s_{12} + s_{21} > t_3 \geq s_{12}$; and Case c where $t_3 < s_{12}$. Figure 9 depicts the Case b scenario.
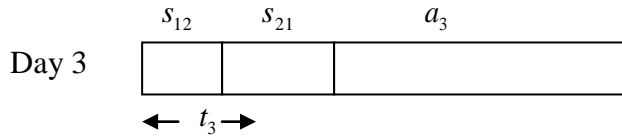

Figure 9: Containers in Terminal on Day 3 (Case b)

Following the same logic as for Day 2, we can determine the number of containers that will be left behind after Day 3's pick-ups, and the specific numbers that are from Day 1, Day 2, and Day 3's arrivals with the formulas given below:

$$s_{13} = \max(0, s_{12} - t_3)$$
$$s_{22} = \max(0, s_{21} - \max(0, t_2 - s_{12}))$$
$$s_{31} = a_3 + s_{12} + s_{21} - t_3 - s_{13} - s_{22}$$
$$= a_3 + (s_{12} - s_{13}) + (s_{21} - s_{22}) - t_3$$

From the above illustrations, we can deduce that on Day n, as a_n containers arrive and t_n containers are trucked away, the numbers remained in the terminal from previous arrivals can be computed as follows:

$$s_{h,n-h+1} = \max(0, s_{h,n-h} - $$

$$\max(0, t_n - \sum_{k=1}^{h-1} s_{k,n-k})),$$

$$\forall h \in \{1, \cdots, n-1\}$$

$$s_{n1} = a_n + \sum_{k=1}^{n-1} (s_{k,n-k} - s_{k,n-k+1}) - t_n$$

On any day, say Day n, containers that incur demurrage charge are those that have been in the terminal for over the allowable free storage duration, F, i.e.

$$p_n = \sum_{k>F} s_{n-k+1,k}$$

Then the demurrage cost on Day n can be obtained simply by multiplying p_n with the per day demurrage charge.

The General Model

Companies often contract multiple carriers for shipping their containers. Each carrier maintains a different voyage schedule and the terminal they use in the destination port may apply different free storage durations for the arriving containers. Moreover, there are different types of containers that companies may ship, i.e., the twenty-foot and the forty-foot marine containers. Shipping charges of the two different types of containers are different. To accommodate these added complexities, we introduce two superscripts in the notation, one for the carrier, and the other for container type, in the following definitions:

a_n^{ij} : Number of containers of Type j arrive on Day n via Carrier i

s_{nk}^{ij} : Number of containers of Type j arrive on Day n via Carrier i that have been kept in the terminal for k days

t_n^{ij} : Number of containers of Type j transported off Carrier i's terminal on Day n

p_n^{ij} : Number of containers of Type j that incur demurrage charge from Carrier i's terminal on Day n

D_n^{j} : Number of containers of Type j the company need to be transported to the warehouse on Day n

K_n^{ij} : Shipping capacity of containers of Type j on the ship of Carrier i that arrives at destination port on Day n

H^{ij} : Shipping charge per container of Type j by Carrier i

F^i: Number of free storage days provided by Carrier i

M^{ij}: Demurrage charge per day per container of Type j by Carrier i's terminal

T^j: Trucking cost per container of Type j from the destination port to the warehouse

C_n^j: Trucking capacity of containers of Type j on Day n

The problem of finding the least end-to-end cost for transporting containers from the point of origin to the warehouse can be modeled as follows:

Minimize the cost function below that accounts for the shipping, demurrage, and trucking costs for all container types via all carriers over a given planning horizon:

$$\sum_{ij}(H^{ij} * \sum_n a_n^{ij}) + \sum_{ij}(M^{ij} * \sum_n p_n^{ij})$$
$$+ \sum_j(T^j * \sum_n \sum_i t_n^{ij})$$

Subject to the following constraints:

1. The arrivals of containers are limited to contracted carrier's scheduled ship arrivals. Specifically, the number of containers that can arrive at the destination port cannot exceed the capacity available on the ship contracted for the shipper. On a day when the carrier has no scheduled ship arriving at the destination port, its capacity on that day is set to zero.

$$a_n^{ij} \le K_n^{ij} \qquad \forall n, i, j$$

2. We must move enough containers to the warehouse to meet the demand of the day. This is true for containers of all types.

$$\sum_{k=1}^n \sum_i t_k^{ij} \ge \sum_{k=1}^n D_k^j \quad \forall n, j$$

3. The number of containers transported is limited by the capacity of the truck service.

$$\sum_i t_n^{ij} \le C_n^j \qquad \forall n, j$$

4. Arriving containers add to the current stock in the terminal. Some from the new arrivals will remain in the terminal for the first day after the required number has been transported out. Priority for transportation is assumed to be given to those that are already in the terminal.

$$s_{n1}^{ij} = a_n^{ij} + \sum_{k=1}^{n-1}(s_{k,n-k}^{ij} - s_{k,n-k+1}^{ij}) - t_n^{ij}, \quad \forall n, i, j$$

5. Some containers from previous days' arrivals may continue to stay in the terminal, depending on the number transported.

$$s_{nh}^{ij} = \max(0, s_{n,h-1}^{ij} -$$
$$\max(0, t_{n+h-1}^{ij} - \sum_{k=1}^{n-1} s_{k,n+h-1-k}^{ij}))$$
$$\forall n, i, j, \text{ and } h \in \{2, \cdots, n-1\}$$

6. On any day, the number of containers that have remained in the terminal beyond the free storage duration will incur demurrage charge.

$$p_n^{ij} = \sum_{k>F^i}^{n} s_{n-k+1,k}^{ij} \qquad \forall n, i, j$$

7. To help keep the model size manageable, we observe that most carriers maintain mostly regular voyage schedules. For example, the ship arrival schedules of both the Maersk and NYK lines at the Port of Los Angeles show a regular interval of 7 days (Maersk 2009, NYK 2009). Clearly the cumulative shipping volume must meet the cumulative demands until the next shipping date, and there is no reason to ship more than what is needed to meet the cumulative demands until the next ship arrival date. However, when a shipper employs multiple carriers that may maintain schedules with possibly different vessel arrival intervals and provide different free container storage allowance, it becomes difficult to define when the next ship arrival should be. The following constraint is defined by considering each carrier's vessel arrival cycle separately. In the formula below, B_i is the first vessel arrival day of Carrier i in the planning horizon, and L_i represents the length of vessel arrival cycle time for Carrier i.

$$\sum_{n=B_i+(k-1)L_i}^{B_i+kL_i-1} a_n^{ij} \leq \sum_{n=B_i+(k-1)L_i}^{B_i+kL_i-1} D_n^j$$

$\forall j$ and $\forall k \geq 1$ such that n is within the planning horizon, It is a common knowledge that suppliers and distributors often want to maintain certain reserve to avoid risking inadequate supply. We assume that such reserve is built in as demands in our model.

The Dole Case

We implemented an example model on the General Algebraic Modeling System (GAMS 2009) using its CONOPT solver (GAMS 2009). In this example scenario Dole Foods transports its containers from Southeast Asia to the Ports of Los Angeles using two different carriers. Table 1 defines model parameters including shipping rates and free dwell time allowance used in the example. We also assume that the demurrage charge is $100 per day per container, and the trucking cost per container is $300. Daily demands are assumed 5 for 20' and 10 for 40' containers.

We ran the model for a one-year planning horizon and compare the optimal cost to a solution that a current practice of a 60-40 percent split of the company's shipments on Carrier 1 versus Carrier 2. The results indicate that a savings of 15% may be achieved by following the optimal shipment plan provided by the model solution.

Table 1. Model Parameters

Carrier	Shipping	Rate	Free Dwell Time	Vessel Arrival
	20'	40'	Allowance (days)	Interval (days)
1	2800	3500	14	7
2	2200	3000	5	7

AnyLogic Simulation

In our next effort we designed and implemented a simulation of the Southern California Agile Supply Network (SCASN). This model, constructed to support SM21 for regional initiatives currently within Southern California, is intentionally designed to be flexible and generic such that it can be applied to any region or agile networking scenario. The overall SCASN model architecture offers numerous benefits for this project and beyond. Among them are:

- A commercial off-the-shelf (COTS) simulation tool (AnyLogic) is used and preferred by USTRANSCOM.
- Generic Implementation and Extensibility. Outline of physical and logical elements defined that can be reconfigured to represent enhanced regional infrastructure, etc.
- Provides a flexible, high-level network oriented view that can be used for existing and SCASN infrastructure improvements.
- It is generic such that it is possible to use this architecture for other regions.
- Any combination of facilities and transportation connections can be explored.
- The approach allows you to look at the entire region and its relationship to sourcing and shipments between nodes outside of the region (CONUS).
- Provides the ability to create a "Regional Network Flow Representation" that synchronizes:
  - ➢ Regional Infrastructure: Facilities, Transportation Links, etc.
  - ➢ Transportation Strategies (freight volume flows through the infrastructure).
  - ➢ Freight Volume Demands: Vessel, Rail, OTR, other arrivals
- The simulation model can be made available as a Web Service.

AnyLogic

Initially we began by implementing the simulation using ARENA. Since, however, we realized that it would be very difficult if not impossible to integrate ARENA into the Service Oriented Architecture (SOA) we looked for alternative simulation engines. The Java-based AnyLogic

simulation tool was found to fit this need. We converted our ARENA based simulation model to an ANYLOGIC based simulation model.

AnyLogic is a dynamic simulation tool that brings together System Dynamics, Process-centric (AKA Discrete Event), and Agent Based approaches within one modeling language and one model development environment. The language of AnyLogic has unmatched flexibility and enables the user to capture the complexity and heterogeneity of business, economy and social systems at any desirable level of details. AnyLogic set of primitives and library objects allows to model manufacturing and logistics, business processes, human resources, consumers' behavior, as well as the environment (the "background") in their natural interaction. The object-oriented model design paradigm supported by AnyLogic provides for modular and incremental construction of large models. AnyLogic 6 is based on Java and the Eclipse framework. Eclipse is an open source community, whose projects are focused on building an open development platform comprised of extensible frameworks, tools and runtimes for building, deploying and managing software across the lifecycle. Since AnyLogic is based on Java, it is ideally suited to be part of an Internet based Service Oriented Architecture (SOA). Applications developed on AnyLogic can be made available as a Web Service using Apache Axis.

Our application as a standalone application currently can be accessed by users as a Web Service. Users are able to start and stop a model remotely and are able to observe the model run and a set of basic statistics through a browser window. We will provide the following future capabilities:

- Allow users to start a simulation run remotely and to upload their own networks.
- Improve the Graphical User Interface for the user.
- We will add capabilities to our model that allow users to study different troop deployment scenarios and their impact on commercial shipping.
- The application will contain example scenarios that enable the user to experiment by changing some of the parameters in these example scenarios.
- The user will be able to modify these scenarios for sensitivity analysis.
- The application will include the capability to make changes to the underlying network for sensitivity analysis and forecasting.
- The application will provide a user interface with intuitive and self explanatory menus for defining a new network and new scenarios. This will provide the user with utmost flexibility with respect to the regions / scenarios the user will be able to analyze.
- Supports different simulation paradigm approaches, whichever is appropriate for the given scenario.

Our AnyLogic model development and model logic are illustrated in Figures 10 and 11. A user has the option to define the logic of the simulation using modules that are included in the AnyLogic simulation package (Figure 10) or define new custom classes as needed (Figure 11).
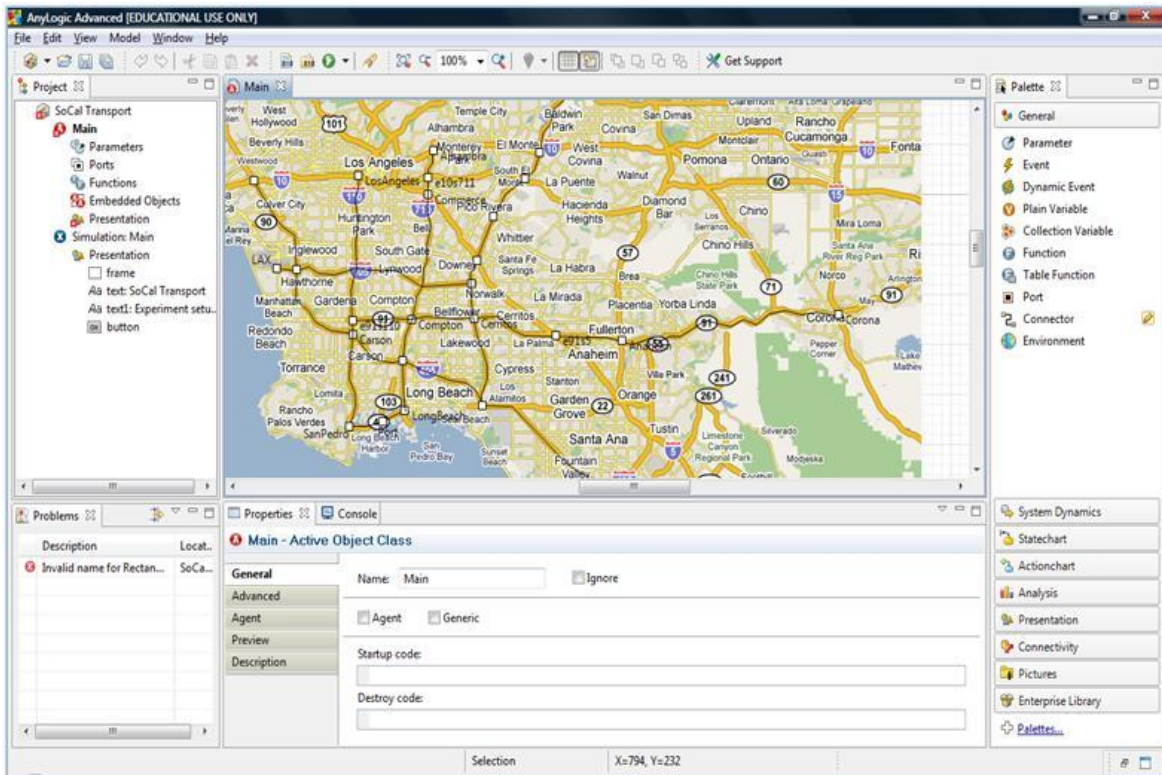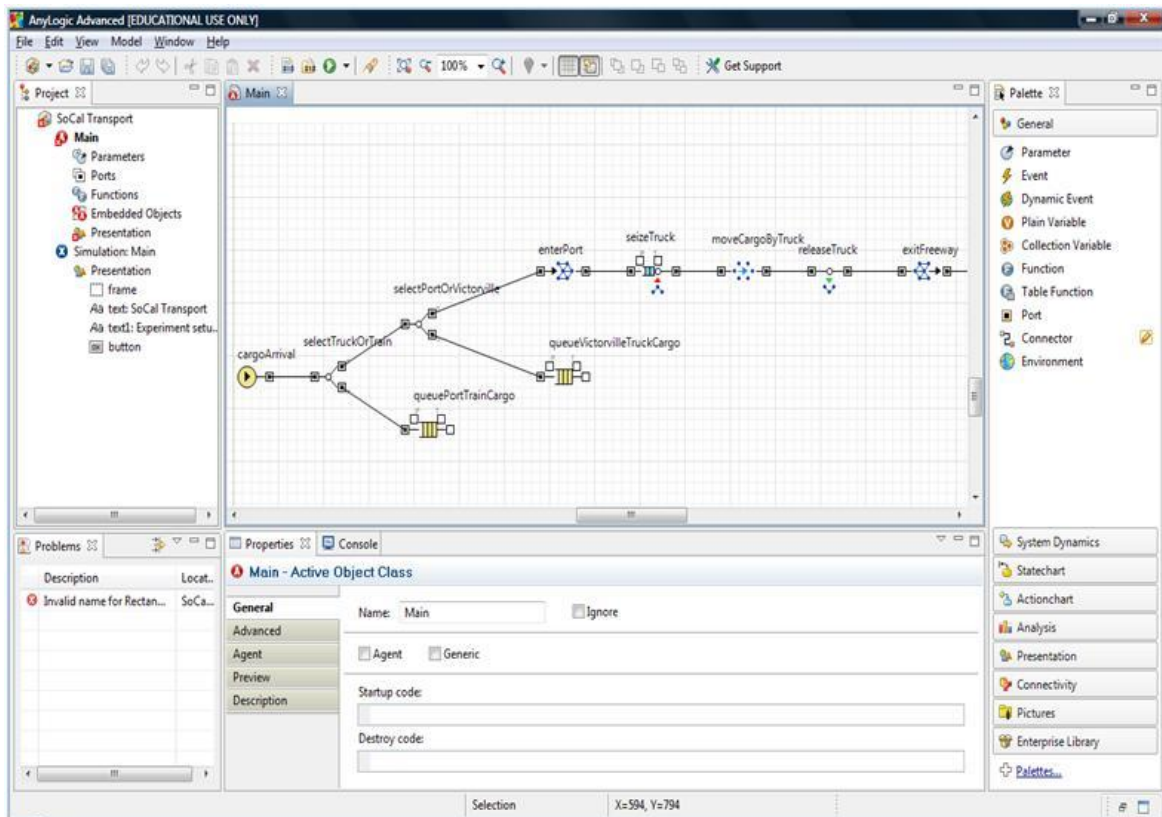
Figure 10: AnyLogic Model Development

Figure 11: AnyLogic Model Logic

Page 31

Web Services

Web Services enable cloud computing by decoupling users from service providers. They are cross-platform cross-language distributed computing applications. In the context of a Web Service a user is called a *consumer*. A Web Services technical interface can be described using an XML document that is readable by any web browser. Moreover the *consumer* of a Web Service does not need to know anything about the Web Services implementation. The consumer only needs to know the Web Service interface i.e. how to interact with it. In essence Web Services provide process integration.
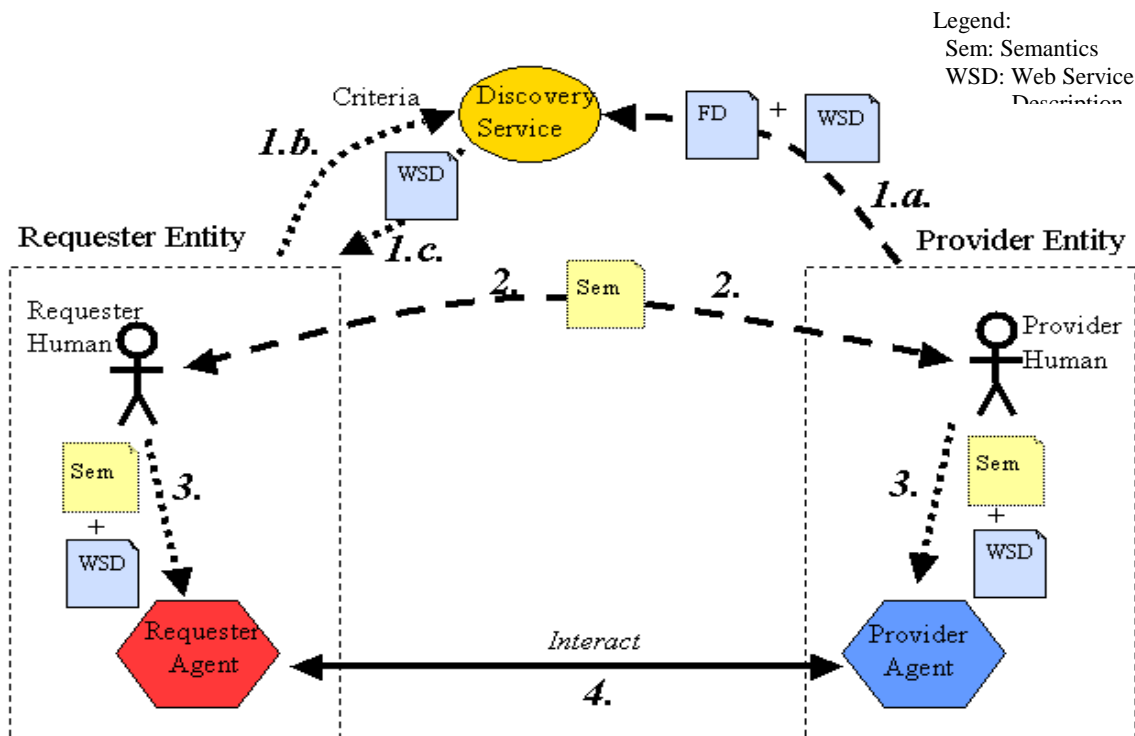

Figure 12: Web Service Illustration

Figure 12 shows the functionality of a Web Service. (1) the requester (consumer) and provider become known to each other (or at least one becomes know to the other); (2) the requester and provider entities agree on the service description and semantics that will determine the interaction between the requester and provider agents; (3) the service description and semantics are implemented by the requester and provider agents; and (4) the requester and provider agents exchange messages, thus performing some task on behalf of the requester and provider entities. (I.e., the exchange of messages with the provider agent represents the concrete manifestation of interacting with the provider entity's Web service.) (http://www.w3.org)

Once implemented a Web Service can be accessed from another application directly (computer to computer without human intervention) or simply through a browser window. The Simple Object Access Protocol (SOAP) represents one viable way to offer web services.

SOAP

SOAP is an XML-based communication protocol and encoding format for inter-application communication. SOAP is widely viewed as the backbone of Web Services. SOAP consists of three parts: an envelope that defines a framework for describing what is in a message and how to process it, a set of encoding rules for expressing instances of application defined data types, and a convention for representing remote procedure calls and responses. We used Axis 2.0 to implement our AnyLogic SOAP-based Web Service.

Axis 2.0

Axis is essentially a SOAP engine, i.e., a framework for constructing SOAP processors such as clients, servers, gateways, etc. Axis comes with security engines that can provide encryption, digital signatures, and timestamps.
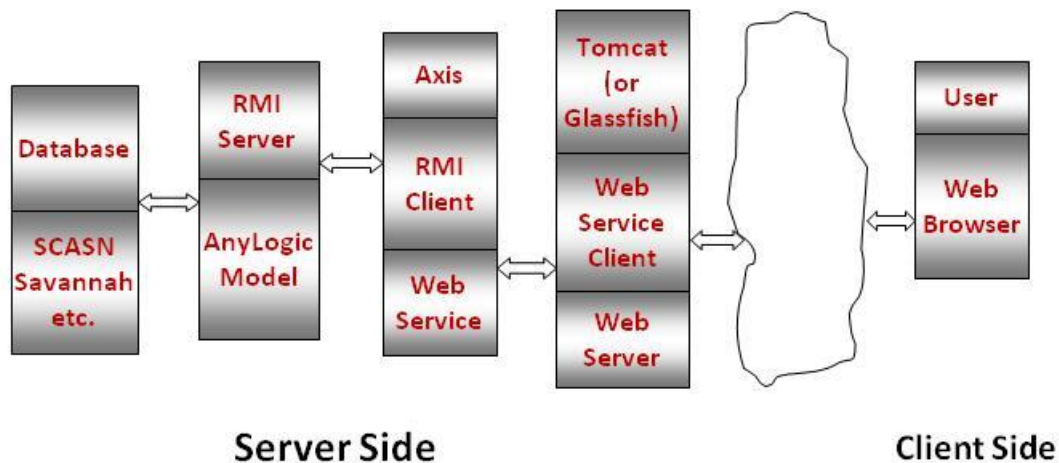
Architecture of the Web Service

Figure 13: AnyLogic Web Service Architecture

Figure 13 shows the architecture of our AnyLogic Web Service. A user interacts through a browser with a web Service Client on the server side. This client is implemented in Glassfish, an open source Web Server. The Web Service Client then interacts with a Web Service implemented through Axis. This service uses a Remote Method Invocation (RMI) Client to interact with an RMI server on which the Java based AnyLogic Model is running. The running model has access to a database on which network and network attributes are stored. Through Web Services it is possible to remotely modify and run an AnyLogic simulation without owning

an AnyLogic license. This will allow us to integrate the simulation capabilities into the Service Oriented Architecture (SOA). Users will be able to define their own networks and data sets, upload them through the Web Service to an AnyLogic server and execute, modify, observe and analyze the resulting running simulation from their browser window again as a Web Service application.

In our current implementation as illustrated in Figure 13, the client side support of the Web Service is placed on the server side to alleviate any need of a software installation on the client side. An alternative architecture will place this support on the client side computer. This architecture will require software download and installation on the client side computer. The advantage of this method is to enable more processing to be performed on the client side, thereby allowing more sophisticated interaction between the client user and the Web Service with faster response.

Current status and future work

Currently our AnyLogic simulation model can be called by a user as a Web Service through a browser window. Figure 14 shows a screenshot of the Web Service user interface. A user can pause and restart a running simulation, observe the simulation run through a window that shows containers moving through the Southern California region and receive some basic statistics about the simulation run, namely the percentage of available trucks currently in use. In the example shown 500 available trucks are assumed.
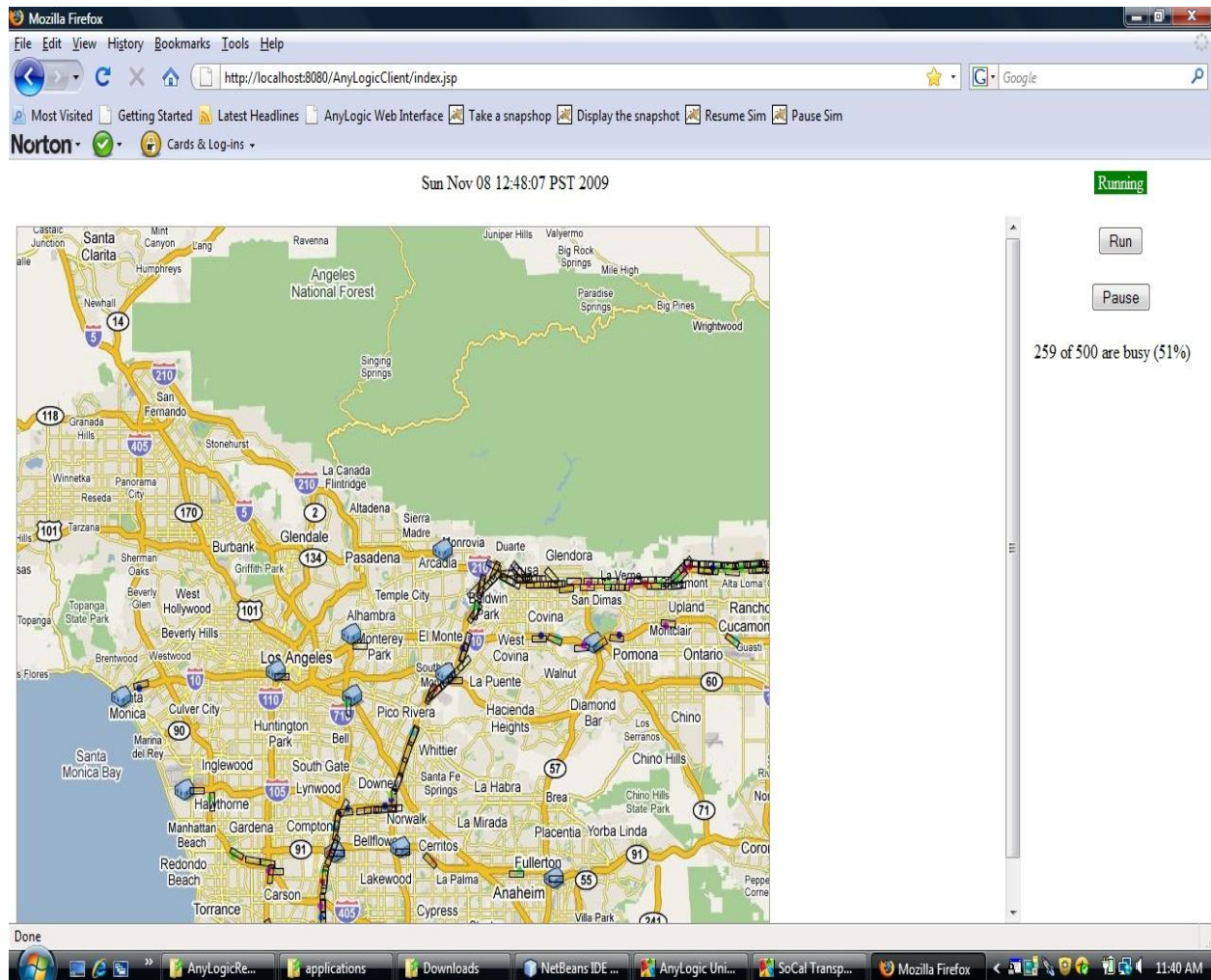
Figure 14: Web Service User Interface

In the shown example the Web Service is accessed through a browser window from local host, simulating remote access. The Web Service implementation, however, is not yet completed. We need to add several functionalities:

- Currently the Web Service only allows a client to pause and restart a simulation run that was started earlier on the server side. Users should be able to upload their own networks to the server and start execution of (their own) simulations remotely.
- Currently the Web Service Graphical User Interface (GUI) is very basic. A screen that shows the simulation run refreshes every 5 seconds. Also only a small subset of statistics is provided to the user. We need to finish the design of the user interface.
- We have developed a tool to accept user input for network definition in the form of Excel spread sheet and convert it into an SQL database, which drives the AnyLogic model. We need to develop a user interface to enable this input to be done remotely as part of the Web Service application.
- We will add capabilities to our model that allow users to study different military deployment scenarios and their impact on commercial shipping.

Page 35

- The application will contain example scenarios that enable the user to experiment by changing some of the parameters in these example scenarios.
- The user will be able to modify these scenarios for sensitivity analysis.
- The application will include the capability to make changes to the underlying network for sensitivity analysis and forecasting.
- The Web Service application must be integrated into the SOA.

Appendix A – Web Service User Manual

Currently the functionality of the AnyLogic Web Service is limited. Figure A1 below shows a screenshot of a browser window through which the AnyLogic Web Service is accessed.
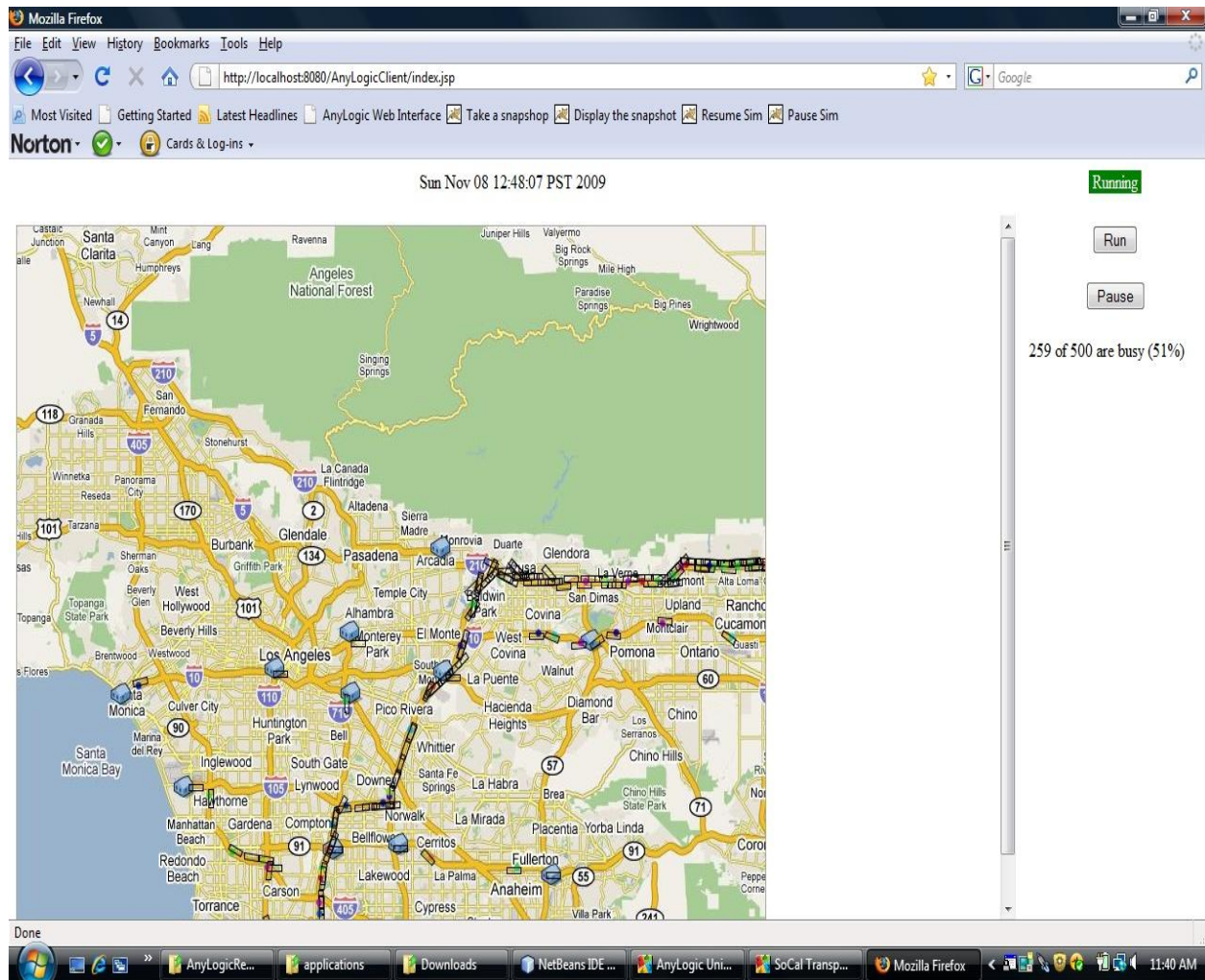

Figure A1 AnyLogic Web Service access through a Web browser

To execute an AnyLogic model as a Web Service an administrator must first start the model execution on the server. Through a Web browser window, a user can remotely observe, pause and restart this model execution.

The model execution is paused by pressing the Pause – button. The model execution is restarted by pressing the Run – button. Below the Pause – button the user is continuously informed about the number of trucks that are in use at this point in time of the model execution. The browser window also contains a graphical depiction of the model execution. In the example In Figure A1 above, trucks a removing as small rectangles on a map of the Southern California Region. The

graphical depiction is refreshed every 5 seconds from the server. The current simulation time is shown on top of the graphical depiction of the model execution.

Appendix B – References

Ablaze Development Corp. *Multi-Modal Terminal Model Documentation.* Version 2, Strategic

      Mobility 21, July 2008.

Strategic Mobility 21. *SCASN Getting Started Checklist*. July 31, 2008.

Strategic Mobility 21. *Southern California Agile Supply Network Simulation Model User's*

      Manual. July 31, 2008.

TranSystems Corporation. *Simulation Analysis Report*. Strategic Mobility 21, July 31, 2008

TranSystems Corporation. *Trade Corridor Gap Analysis.* Strategic Mobility 21, July 31, 2008

World Wide Web Consortium. *World Wide Web Consortium Homepage*. www.w3.org.

Appendix C – List of Acronyms

Acronym	Term
APS	Agile Port System
ASN	Agile Supply Network
CCDoTT	Center for the Commercial Deployment of Transportation Technologies
CONUS	Continental United States
COTS	Commercial Off-the-Shelf
DC	Distribution Center
FIFO	First-In-First-Out
GAMS	General Algebraic Modeling System
GDX	GAMS Data Exchange
GTMS	Global Transportation Management System
GUI	Graphical User Interface
JDDE	Joint Deployment Distribution Enterprise
JDDSP	Joint Deployment Distribution Support Platform
MMT	Multi-Modal Terminal
MTM	Multi-Modal Terminal Model
MSA	Modeling, Simulation, and Analysis
RMI	Remote Method Invocation
SCASN	Southern California Agile Supply Network
SCLA	Southern California Los Angeles
SM21	Strategic Mobility 21
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
TSP	Traveling Salesman Problem
USTRANSCOM	United States Transportation Command
YTD	Year to Date